



# An efficient algorithm for constructing Bayesian optimal choice designs

Roselinde Kessels, Bradley Jones, Peter Goos and Martina Vandebroek

DEPARTMENT OF DECISION SCIENCES AND INFORMATION MANAGEMENT (KBI)

# An efficient algorithm for constructing Bayesian optimal choice designs

Roselinde Kessels\*  
Bradley Jones<sup>†</sup>  
Peter Goos<sup>‡</sup>  
Martina Vandebroek<sup>§</sup>

## Abstract

Recently, Kessels et al. (2006) developed a way to produce Bayesian  $\mathcal{G}$ - and  $\mathcal{V}$ -optimal designs for the multinomial logit model. These designs allow for precise response predictions which is the goal of conjoint choice experiments. The authors showed that the  $\mathcal{G}$ - and  $\mathcal{V}$ -optimality criteria outperform the  $\mathcal{D}$ - and  $\mathcal{A}$ -optimality criteria for prediction. However, their  $\mathcal{G}$ - and  $\mathcal{V}$ -optimal design algorithm is computationally intensive, which is a barrier to its use in practice. In this paper, we present an efficient algorithm for calculating Bayesian optimal designs by means of the different criteria. Particularly, the speed of computation for the  $\mathcal{V}$ -optimality criterion has improved dramatically. The new algorithm makes it possible to use Bayesian  $\mathcal{D}$ -,  $\mathcal{A}$ -,  $\mathcal{G}$ - and  $\mathcal{V}$ -optimal designs that are tailored to individual respondents in computerized conjoint choice studies.

*Keywords:* conjoint choice design, multinomial logit, Bayesian  $\mathcal{D}$ -,  $\mathcal{A}$ -,  $\mathcal{G}$ - and  $\mathcal{V}$ -optimality, adaptive algorithm, Cholesky decomposition, coordinate-exchange

## 1 Introduction

Conjoint choice experiments or more succinctly, choice experiments, are widely used in marketing to measure how the attributes of a product or service jointly affect consumer preferences. In a choice experiment, a product or service is characterized by a combination of attribute levels called a profile or an alternative. Respondents then choose one from a group of profiles called a choice set. They repeat this task for several other choice sets presented to them. All submitted choice sets make up the experimental design. The

---

\*Department of Decision Sciences and Information Management, Faculty of Economics and Applied Economics, Katholieke Universiteit Leuven, Naamsestraat 69, 3000 Leuven, Belgium.

<sup>†</sup>SAS Institute Inc., SAS Campus Drive, Cary, NC 27513, USA.

<sup>‡</sup>Department of Mathematics, Statistics and Actuarial Sciences, Faculty of Applied Economics, Universiteit Antwerpen, Prinsstraat 13, 2000 Antwerpen, Belgium.

<sup>§</sup>Department of Decision Sciences and Information Management & University Center for Statistics, Katholieke Universiteit Leuven, 3000 Leuven, Belgium.

aim of a choice experiment is to estimate the importance of each attribute and its levels based on the respondents' preferences. The estimates are then exploited to mimic real marketplace choices by making predictions about consumers' future purchasing behavior.

Designing an efficient choice experiment involves selecting those choice sets that result in an accurately estimated model providing precise predictions. Kessels et al. (2006) compared four different design criteria based on the multinomial logit model to reach this goal. They studied the predictive performance of the  $\mathcal{D}$ - and  $\mathcal{A}$ -optimality criteria versus the  $\mathcal{G}$ - and  $\mathcal{V}$ -optimality criteria. Special attention was paid to the  $\mathcal{G}$ - and  $\mathcal{V}$ -optimality criteria which aim at making precise predictions. The authors were the first to work out these criteria for the multinomial logit model. On the other hand, the  $\mathcal{D}$ - and  $\mathcal{A}$ -optimality criteria focus on accurate estimates. Until now, the  $\mathcal{D}$ -optimality criterion has been most often employed to construct efficient choice designs (see Huber and Zwerina 1996; Sándor and Wedel 2001).

Because the multinomial logit model is nonlinear in the parameters, the computation of the optimality criteria depends on the unknown parameter vector. To solve this problem, Kessels et al. (2006) adopted a Bayesian design procedure as proposed by Sándor and Wedel (2001). Following these authors, they approximated the design criteria using a Monte Carlo sample from a multivariate normal prior parameter distribution. Monte Carlo sampling involves taking a large number of random draws from a probability distribution as a surrogate for that distribution. Like Sándor and Wedel (2001), Kessels et al. (2006) used 1,000 random draws. The four optimality criteria in the Bayesian context are labelled the  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimality criteria. Kessels et al. (2006) implemented these criteria in a modified Fedorov algorithm (Cook and Nachtsheim 1980; Fedorov 1972) to construct  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs. We refer to their complete algorithm as the Monte Carlo modified Fedorov algorithm (MCMF).

Kessels et al. (2006) showed that the  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimality criteria outperform the  $\mathcal{D}_B$ - and  $\mathcal{A}_B$ -optimality criteria in terms of prediction accuracy. They warn, however, that the computation of  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs is substantially more demanding than the search for  $\mathcal{D}_B$ - and  $\mathcal{A}_B$ -optimal designs. The long computing times resulting from MCMF make the  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimality criteria impractical to use. Also, the computational burden implies that the application of the  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimality criteria to computerized conjoint choice studies is limited. Ideally, computerized conjoint studies use choice designs that are tailored to the individual respondents so that maximum information is obtained on the individuals' preferences and thus on the heterogeneity between subjects.

The goal of this paper is to present a novel design construction algorithm that is much faster than MCMF employed by Kessels et al. (2006). The speed of the new algorithm allows the  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimality criteria to be used in practice and it also opens the perspective of applying individualized Bayesian optimal choice designs in web-based conjoint studies.

The new algorithm has four key features. First, it uses an update formula to economically calculate the change in any of the optimality criteria for two designs that differ only in one profile. In this way, the optimality criterion values do not need to be re-computed from scratch. Second, it involves a formula for the  $\mathcal{V}_B$ -optimality criterion so that its computation is even more efficient. Third, the algorithm is an adaptation of Meyer and Nachtsheim's (1995) coordinate-exchange algorithm which is much faster than the modified Fedorov algorithm. Lastly, it relies on a designed sample of only 20 prior parameters instead of the Monte Carlo sample of 1,000 draws. However, the algorithm still checks the designs produced by each random start using the Monte Carlo sample. Because of this re-evaluation, the algorithm is called the adaptive algorithm.

The outline of the remainder of the paper is as follows. Section 2 reviews the  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimality criteria for the multinomial logit model. In Section 3, we present the adaptive algorithm as an alternative to MCMF for faster computation of the optimal designs for all four criteria. We use the design example from Kessels et al. (2006) for comparison purposes. Section 4 discusses the four key features of the adaptive algorithm and Section 5 considers a more challenging scenario made possible by the faster method. Section 6 summarizes the results and suggests some opportunities for further research.

## 2 Design criteria for the multinomial logit

To present our improved design construction approach, we start with an overview of the different design criteria for the multinomial logit model. The model draws on a choice design matrix  $\mathbf{X} = [\mathbf{x}'_{js}]_{j=1,\dots,J;s=1,\dots,S}$  where  $\mathbf{x}_{js}$  is a  $k \times 1$  vector of the attribute levels of profile  $j$  in choice set  $s$ . A respondent's utility for that profile is modelled as  $u_{js} = \mathbf{x}'_{js}\boldsymbol{\beta} + \varepsilon_{js}$  where  $\boldsymbol{\beta}$  is a  $k \times 1$  vector of parameters and  $\varepsilon_{js}$  is an i.i.d. extreme value error term. The multinomial logit probability a respondent chooses profile  $j$  in choice set  $s$  is  $p_{js} = e^{\mathbf{x}'_{js}\boldsymbol{\beta}} / \sum_{t=1}^J e^{\mathbf{x}'_{ts}\boldsymbol{\beta}}$ . The information matrix  $\mathbf{M}$ , which is the inverse of the variance-covariance matrix of the parameter estimators, is the sum of the information matrices of the  $S$  choice sets  $\mathbf{M}_s$  as shown below:

$$\begin{aligned} \mathbf{M}(\mathbf{X}, \boldsymbol{\beta}) &= N \sum_{s=1}^S \mathbf{M}_s(\mathbf{X}_s, \boldsymbol{\beta}) \\ &= N \sum_{s=1}^S \mathbf{X}'_s (\mathbf{P}_s - \mathbf{p}_s \mathbf{p}'_s) \mathbf{X}_s, \end{aligned} \tag{1}$$

where  $\mathbf{X}_s = [\mathbf{x}_{1s}, \dots, \mathbf{x}_{Js}]'$ ,  $\mathbf{p}_s = [p_{1s}, \dots, p_{Js}]'$ ,  $\mathbf{P}_s = \text{diag}[p_{1s}, \dots, p_{Js}]$  and  $N$  is the number of respondents. Kessels et al. (2006) implemented different design criteria or functions of the information matrix (1) for constructing optimal choice designs. This task is complicated by the fact that the information on the parameters depends on the unknown values of those parameters through the probabilities. Therefore, the authors adopted a Bayesian design strategy that integrates the design criteria over a prior parameter distribution

$\pi(\boldsymbol{\beta})$ . The multivariate normal distribution  $\mathcal{N}(\boldsymbol{\beta}|\boldsymbol{\beta}_0, \boldsymbol{\Sigma}_0)$  was chosen for this purpose.

The design criteria employed are the  $\mathcal{D}$ -,  $\mathcal{A}$ -,  $\mathcal{G}$ - and  $\mathcal{V}$ -optimality criteria. The  $\mathcal{D}$ - and  $\mathcal{A}$ -optimality criteria both are concerned with a precise estimation of the parameters  $\boldsymbol{\beta}$  in the multinomial logit model. The  $\mathcal{D}$ -optimality criterion aims at designs that minimize the determinant of the variance-covariance matrix of the parameter estimators, while the  $\mathcal{A}$ -optimality criterion aims at designs that minimize the trace of the variance-covariance matrix. The Bayesian  $\mathcal{D}$ -optimality criterion is

$$\mathcal{D}_B = \int_{\mathcal{R}^k} \{\det(\mathbf{M}^{-1}(\mathbf{X}, \boldsymbol{\beta}))\}^{1/k} \pi(\boldsymbol{\beta}) d\boldsymbol{\beta}, \quad (2)$$

with the  $\mathcal{D}_B$ -optimal design minimizing (2). The  $\mathcal{A}_B$ -optimal design minimizes

$$\mathcal{A}_B = \int_{\mathcal{R}^k} \text{tr}(\mathbf{M}^{-1}(\mathbf{X}, \boldsymbol{\beta})) \pi(\boldsymbol{\beta}) d\boldsymbol{\beta}. \quad (3)$$

The  $\mathcal{G}$ - and  $\mathcal{V}$ -optimality criteria were developed to make precise response predictions. These criteria are important in this context since predicting consumer responses is the goal of choice experiments. The  $\mathcal{G}$ - and  $\mathcal{V}$ -optimality criteria for the multinomial logit model were first elaborated by Kessels et al. (2006). They are defined with respect to a design region  $\chi$  consisting of all  $Q$  possible choice sets of size  $J$  that can be composed from the candidate profiles:  $\chi = \{\{\mathbf{x}_{1q}, \dots, \mathbf{x}_{Jq}\} | q = 1, \dots, Q\}$ . A  $\mathcal{G}$ -optimal design minimizes the maximum prediction variance over the design region  $\chi$ , while a  $\mathcal{V}$ -optimality design minimizes the average prediction variance over this region. Formally, the  $\mathcal{G}_B$ -optimality criterion is

$$\mathcal{G}_B = \int_{\mathcal{R}^k} \max_{\mathbf{x}_{jq} \in \chi} \text{var}\{\hat{p}_{jq}(\mathbf{x}_{jq}, \boldsymbol{\beta})\} \pi(\boldsymbol{\beta}) d\boldsymbol{\beta} = \int_{\mathcal{R}^k} \max_{\mathbf{x}_{jq} \in \chi} \mathbf{c}'(\mathbf{x}_{jq}) \mathbf{M}^{-1}(\mathbf{X}, \boldsymbol{\beta}) \mathbf{c}(\mathbf{x}_{jq}) \pi(\boldsymbol{\beta}) d\boldsymbol{\beta}, \quad (4)$$

where  $\hat{p}_{jq}(\mathbf{x}_{jq}, \boldsymbol{\beta})$  denotes the predicted choice probability for  $\mathbf{x}_{jq}$  and

$$\mathbf{c}(\mathbf{x}_{jq}) = \frac{\partial p_{jq}(\mathbf{x}_{jq}, \boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = p_{jq} \left( \mathbf{x}_{jq} - \sum_{t=1}^J p_{tq} \mathbf{x}_{tq} \right), \quad (5)$$

the partial derivative of the multinomial logit probability with respect to  $\boldsymbol{\beta}$ . The  $\mathcal{V}_B$ -optimality criterion is

$$\mathcal{V}_B = \int_{\mathcal{R}^k} \int_{\chi} \mathbf{c}'(\mathbf{x}_{jq}) \mathbf{M}^{-1}(\mathbf{X}, \boldsymbol{\beta}) \mathbf{c}(\mathbf{x}_{jq}) d\mathbf{x}_{jq} \pi(\boldsymbol{\beta}) d\boldsymbol{\beta} \quad (6)$$

with  $\mathbf{c}(\mathbf{x}_{jq})$  given by (5).

### 3 The adaptive algorithm versus MCMF for computing $\mathcal{D}_B$ -, $\mathcal{A}_B$ -, $\mathcal{G}_B$ - and $\mathcal{V}_B$ -optimal designs

We propose the adaptive algorithm for generating  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs instead of the Monte Carlo modified Fedorov algorithm (MCMF) employed by Kessels et

al. (2006) (see Section 1). The adaptive algorithm is much faster than MCMF so that for a given computing time the resulting designs outperform the designs produced by MCMF.

We illustrate the better results from the adaptive algorithm versus MCMF using the design example of Kessels et al. (2006). These authors constructed  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs of two classes:  $3^2 \times 2/2/12$  and  $3^2 \times 2/3/8$ . The design profiles in the two classes have a similar attribute structure with two attributes at three levels and one attribute at two levels. Hence, the sets of candidate profiles of the classes comprise the same  $3^2 \times 2 = 18$  profiles. The designs of the first class consist of twelve choice sets of size two, while the designs of the second class consist of eight choice sets of size three. So, the designs of both classes contain 24 profiles. Since we exploit this design example of 24 profiles to compare the adaptive algorithm to MCMF, we refer to it as the comparison example and label it  $3^2 \times 2/24$ .

Using effects-type coding (see Kessels et al. 2006), the number of elements,  $k$ , in the parameter vector is five. As prior parameter distribution, Kessels et al. (2006) proposed the multivariate normal distribution  $\pi(\boldsymbol{\beta}) = \mathcal{N}(\boldsymbol{\beta}|\boldsymbol{\beta}_0, \boldsymbol{\Sigma}_0)$ , with  $\boldsymbol{\beta}_0 = [-1, 0, -1, 0, -1]'$  and  $\boldsymbol{\Sigma}_0 = \mathbf{I}_5$ . They approximated this distribution by drawing a Monte Carlo sample of  $R = 1,000$  prior parameter values  $\boldsymbol{\beta}^r$ ,  $r = 1, \dots, R$ , from it. The Bayesian optimal designs were then obtained from 200 tries or random starts of the modified Fedorov algorithm. This algorithm iteratively improves the starting design by exchanging its profiles with profiles from the candidate set. To compute the  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimality criteria for the two-alternative designs, the design region  $\chi$  consists of  $Q = \binom{18}{2} = 153$  choice sets or 306 profiles, whereas for the three-alternative designs, it includes  $Q = \binom{18}{3} = 816$  choice sets or 2,448 profiles.

Based on the same normal prior distribution we employed the adaptive algorithm to reproduce the  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs for the comparison example. Besides the two- and three-alternative designs, we also generated the four-alternative designs containing six choice sets. The design region  $\chi$  in this case is quite extensive involving  $Q = \binom{18}{4} = 3,060$  choice sets or 12,240 profiles. The optimal designs from the adaptive algorithm appear in Tables A.1, A.2 and A.3 of the Appendix. In Table 1, we compared their criterion values with the criterion values from MCMF that we copied from the work of Kessels et al. (2006). As can be seen, the two-alternative  $\mathcal{D}_B$ -optimal designs from both algorithms are equivalent. However, in all the other cases with two and three alternatives, the designs generated with the adaptive algorithm outperform the designs generated with MCMF.

The best criterion values from the adaptive algorithm were the result of 1,000 random starts rather than the 200 random starts utilized to obtain the best criterion values from MCMF. Because the adaptive algorithm is so much faster than MCMF, the extra random starts were still accomplished using far less computing time. The computing times for one try of the adaptive algorithm and MCMF appear in Tables 2a and 2b, respectively. We performed all computations in MATLAB 7 using a Dell personal computer with a 1.60 GHz Intel Processor and 2 GB RAM.



Table 1:  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -criterion values of the  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs for the comparison example  $3^2 \times 2/24$  computed using the adaptive algorithm and the Monte Carlo modified Fedorov algorithm.

Optimal design	2 alternatives		3 alternatives		4 alternatives	
	Adaptive	MCMF	Adaptive	MCMF	Adaptive	MCMF
$\mathcal{D}_B$	0.73024	0.73024	0.75362	0.76617	0.86782	—
$\mathcal{A}_B$	6.55212	6.60563	5.97903	6.02261	6.57135	—
$\mathcal{G}_B$	0.49887	0.51997	0.51051	0.51843	0.60494	—
$\mathcal{V}_B$	0.07184	0.07219	0.06267	0.06285	0.05728	—

Tables 2a and 2b show the huge reductions in computing time using the adaptive algorithm. Particularly important are the reductions in computing time for the  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimality criteria. With the adaptive algorithm the construction of the  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs has become practically feasible. Even the four-alternative  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs were generated quickly, while their computation was not doable with MCMF. Notice also the faster running time for the  $\mathcal{V}_B$ -optimality criterion compared with the  $\mathcal{G}_B$ -optimality criterion. This is due to a computational short cut in the calculation of the  $\mathcal{V}_B$ -optimality criterion which we lay out in Section 4.2.

Table 2: Computing times for one try of the adaptive algorithm and the Monte Carlo modified Fedorov algorithm to generate the Bayesian optimal designs for the comparison example  $3^2 \times 2/24$ . The times are expressed in hours:minutes:seconds.

a) Adaptive algorithm				b) Monte Carlo modified Fedorov			
Design criterion	# Alternatives			Design criterion	# Alternatives		
	2	3	4		2	3	4
$\mathcal{D}_B$	00:00:03	00:00:04	00:00:05	$\mathcal{D}_B$	00:08:00	00:08:00	—
$\mathcal{A}_B$	00:00:03	00:00:04	00:00:05	$\mathcal{A}_B$	00:08:00	00:08:00	—
$\mathcal{G}_B$	00:00:07	00:00:32	00:04:23	$\mathcal{G}_B$	03:00:00	12:00:00	—
$\mathcal{V}_B$	00:00:03	00:00:05	00:00:08	$\mathcal{V}_B$	03:00:00	12:00:00	—

Note that the adaptive algorithm is computationally less effective per number of tries than MCMF. This can be seen from the plots in Figure 1 in which we compare the estimated expected efficiencies against various numbers of tries of the adaptive algorithm and MCMF for computing the two-alternative  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs. These are the efficiencies to expect if a number of tries are performed with each of the algorithms. Details on the calculation of the expected efficiency from a number of tries can be found in the work of Kessels et al. (2006).

The plots for the two-alternative  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs in Figure 1 are also representative of the two-alternative  $\mathcal{A}_B$ - and  $\mathcal{G}_B$ -optimal designs, respectively. The plots for the three-alternative designs exhibit a similar pattern. From the plots, we observe that the differences in efficiency in favor of MCMF are smaller when a prediction-based design criterion is used instead of an estimation-based design criterion. This might be due to the fact that design optimization with the  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimality criteria is generally more difficult than with the  $\mathcal{D}_B$ - and  $\mathcal{A}_B$ -optimality criteria.

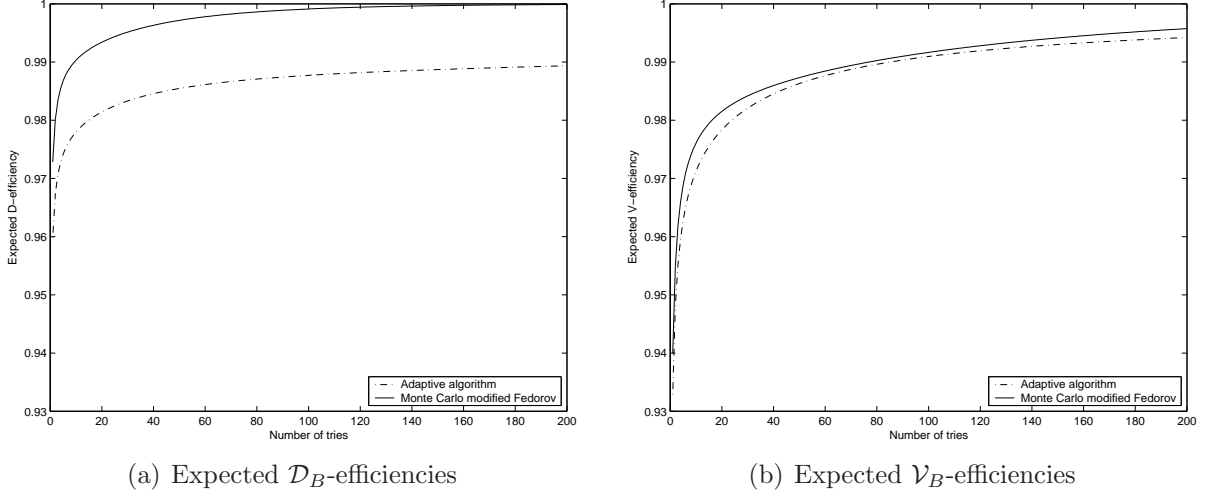


Figure 1: Estimated expected efficiencies against various numbers of tries of the adaptive algorithm and the Monte Carlo modified Fedorov algorithm for computing the two-alternative  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs.

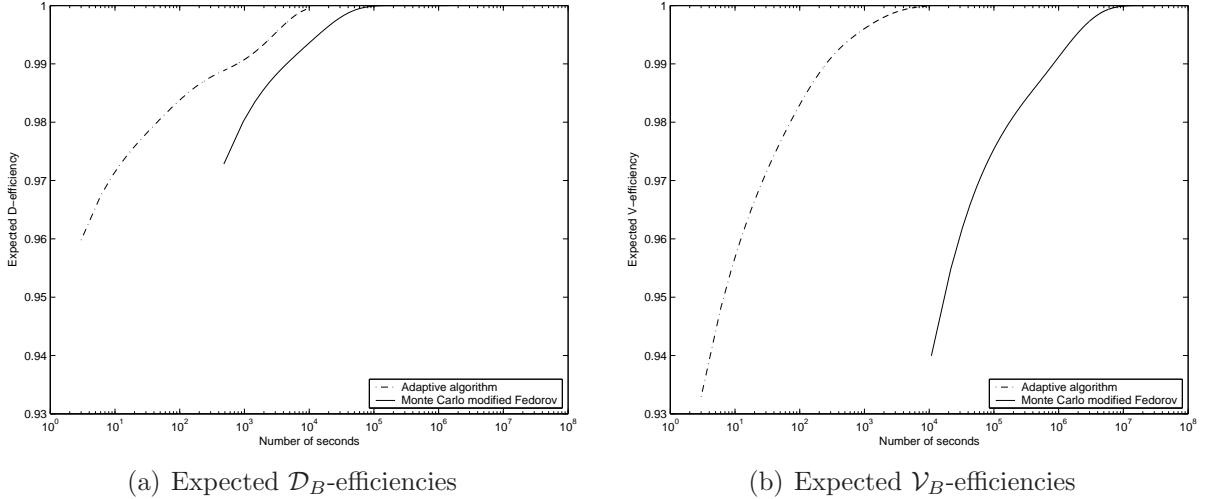


Figure 2: Estimated expected efficiencies against various numbers of seconds of the adaptive algorithm and the Monte Carlo modified Fedorov algorithm for computing the two-alternative  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs.



A more realistic comparison of the effectiveness of the adaptive algorithm versus MCMF appears in the plots of Figure 2. In these graphs, we plotted the estimated expected efficiencies of the two-alternative  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs against the number of seconds of computing time. We expressed the number of seconds on a log-scale. These plots provide compelling evidence of the practical value of the adaptive algorithm. The huge increase in speed created by the adaptive algorithm overtly leads to more efficient designs in a given amount of computing time. This is especially the case for the prediction-based design criteria as illustrated by the plot for the  $\mathcal{V}_B$ -efficiencies. Note however, that the bend in the plot for the  $\mathcal{D}_B$ -efficiencies reveals that the adaptive algorithm has a little difficulty making the final jump from 99% efficiency to 100% or global efficiency.

## 4 Features of the adaptive algorithm

There are four features of the adaptive algorithm that result in increased speed compared with MCMF. They are:

1. updating the Cholesky decomposition of the information matrix,
2. an efficient computation of the  $\mathcal{V}_B$ -optimality criterion,
3. a coordinate-exchange algorithm,
4. a small designed sample of prior parameters.

The next sections discuss each of these in succession.

### 4.1 Updating the Cholesky decomposition of the information matrix

Updating the Cholesky decomposition of the information matrix is an economical way to compute the  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -criterion values of designs that differ only in one profile from another design. The Cholesky decomposition forms a symmetric positive definite matrix as an upper triangular matrix multiplied on the left by its transpose. The information matrix  $\mathbf{M}$  is symmetric because the information matrices of the  $S$  choice sets  $\mathbf{M}_s$  are symmetric. They are of the form  $\mathbf{X}'_s \mathbf{C}_s \mathbf{X}_s$ , where  $\mathbf{C}_s = \mathbf{P}_s - \mathbf{p}_s \mathbf{p}'_s$  is symmetric. If  $\mathbf{M}$  is positive definite, then its Cholesky decomposition is defined as

$$\mathbf{M} = \mathbf{L}'\mathbf{L}, \tag{7}$$

where  $\mathbf{L}$  is an upper triangular matrix named the Cholesky factor.

In the adaptive algorithm, different designs are generated by changing only one attribute level of a single profile at a time (see Section 4.3). The starting design is denoted by  $\mathbf{X}^s$ . We compute the  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -criterion values of each of the designs

as follows. For each prior parameter vector, we compute the information matrix  $\mathbf{M}^s$  through (1) and derive its Cholesky factor  $\mathbf{L}^s$ . We then update the Cholesky factor after every profile change with low rank updates based on the work of Bennett (1965). Using the Cholesky factors the four criterion values for each design can be obtained as shown below. In this way, we avoid re-computation of the information matrix through (1). For the comparison example  $3^2 \times 2/24$ , this procedure reduced the computing times by roughly a factor of three.

We now illustrate how the different design criteria rely on the Cholesky factor  $\mathbf{L}$  of the information matrix  $\mathbf{M}$ . For any vector of coefficients, the  $\mathcal{D}$ -optimality criterion becomes

$$\mathcal{D} = (\det(\mathbf{M}^{-1}))^{1/k} = 1/(\det(\mathbf{M}))^{1/k} = 1/(\det(\mathbf{L}')\det(\mathbf{L}))^{1/k} = 1 / \left( \prod_{i=1}^k l_{ii} \right)^{2/k}, \quad (8)$$

where  $l_{ii}$  is the  $i$ th diagonal element of  $\mathbf{L}$ . Thus, to obtain the  $\mathcal{D}_B$ -criterion value of a design in which a profile has been changed, we do not need to re-compute the information matrix for every prior parameter vector. Only an update of the Cholesky factor is required.

To show the dependency of the  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimality criteria on the Cholesky factor  $\mathbf{L}$ , the Cholesky decomposition (7) has to be inverted. Denoting  $\mathbf{L}^{-1}$  by  $\mathbf{L}_{inv}$ , the inverse is given by

$$\mathbf{M}^{-1} = (\mathbf{L}'\mathbf{L})^{-1} = \mathbf{L}_{inv}\mathbf{L}'_{inv}. \quad (9)$$

Because the Cholesky factor,  $\mathbf{L}$ , is triangular, inverting it is easier than inverting  $\mathbf{M}$ . Then, for any prior parameter vector, the  $\mathcal{A}$ -optimality criterion is

$$\mathcal{A} = \text{tr}(\mathbf{M}^{-1}) = \text{tr}(\mathbf{L}_{inv}\mathbf{L}'_{inv}) = \sum_{i=1}^k \sum_{j=i}^k m_{ij}^2, \quad (10)$$

where  $m_{ij}$  is the  $ij$ th element in  $\mathbf{L}_{inv}$ . So to obtain the  $\mathcal{A}_B$ -criterion value of a design in which a profile has been changed, we need to derive the new Cholesky factor for every prior parameter vector and take its inverse. This goes much faster than computing the new information matrix and inverting it.

In a similar manner, the  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -criterion values are obtained. The prediction variance of profile  $\mathbf{x}_{jq} \in \chi$  is expressed as

$$\mathbf{c}'(\mathbf{x}_{jq})\mathbf{M}^{-1}\mathbf{c}(\mathbf{x}_{jq}) = \mathbf{c}'(\mathbf{x}_{jq})\mathbf{L}_{inv}\mathbf{L}'_{inv}\mathbf{c}(\mathbf{x}_{jq}). \quad (11)$$

Here,  $\mathbf{c}(\mathbf{x}_{jq})$  does not depend on the design  $\mathbf{X}$  and therefore only needs to be computed once for each prior parameter vector. The  $\mathcal{G}_B$ -criterion value is obtained by inserting (11) in (4). For the  $\mathcal{V}_B$ -optimality criterion, we performed some initial calculations that make its computation even more efficient. We describe these calculations in the next section.

## 4.2 Efficient computation of the $\mathcal{V}_B$ -optimality criterion

In the adaptive algorithm, the  $\mathcal{V}_B$ -optimality criterion is implemented in an efficient way. For each prior vector of coefficients, it is possible to compute the average prediction variance without first computing the prediction variances for each profile  $\mathbf{x}_{jq} \in \chi$  separately. A similar approach does not apply to the  $\mathcal{G}_B$ -optimality criterion since finding the worst prediction variance requires the computation of all variances.

To explain our method, we start from the prediction variance (11), but for the sake of clarity, we leave the implementation of the inverse of the Cholesky decomposition for the end. The prediction variance is naturally a scalar since  $\mathbf{c}(\mathbf{x}_{jq})$  is a  $k \times 1$  vector and  $\mathbf{M}^{-1}$  a  $k \times k$  matrix. The trace of a scalar is the scalar itself so that

$$\mathbf{c}'(\mathbf{x}_{jq})\mathbf{M}^{-1}\mathbf{c}(\mathbf{x}_{jq}) = \text{tr}(\mathbf{c}'(\mathbf{x}_{jq})\mathbf{M}^{-1}\mathbf{c}(\mathbf{x}_{jq})). \quad (12)$$

Now,  $\text{tr}(\mathbf{ABC}) = \text{tr}(\mathbf{CAB})$  if  $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$  are matrices such that  $\mathbf{ABC}$  is a square matrix and the matrix product  $\mathbf{CAB}$  exists. This equality is known as the cyclic property of the trace. Since the prediction variance is a scalar and  $\mathbf{c}(\mathbf{x}_{jq})\mathbf{c}'(\mathbf{x}_{jq})$  is a  $k \times k$  matrix that conforms with  $\mathbf{M}^{-1}$ ,

$$\text{tr}(\mathbf{c}'(\mathbf{x}_{jq})\mathbf{M}^{-1}\mathbf{c}(\mathbf{x}_{jq})) = \text{tr}(\mathbf{c}(\mathbf{x}_{jq})\mathbf{c}'(\mathbf{x}_{jq})\mathbf{M}^{-1}). \quad (13)$$

Let  $\mathbf{W}_{jq} = \mathbf{c}(\mathbf{x}_{jq})\mathbf{c}'(\mathbf{x}_{jq})$ . Because  $\mathbf{c}(\mathbf{x}_{jq})$  does not depend on the design  $\mathbf{X}$ ,  $\mathbf{W}_{jq}$  is not a function of  $\mathbf{X}$  either so that it only has to be computed once for each prior parameter vector. We now average the individual matrices  $\mathbf{W}_{jq}$  over all profiles  $\mathbf{x}_{jq} \in \chi$  and denote the subsequent matrix by  $\mathbf{W}$ :

$$\mathbf{W} = \frac{1}{JQ} \sum_{j=1}^J \sum_{q=1}^Q \mathbf{W}_{jq}. \quad (14)$$

The average prediction variance across all profiles  $\mathbf{x}_{jq} \in \chi$  for a given prior parameter vector is then

$$\int_{\chi} \mathbf{c}'(\mathbf{x}_{jq})\mathbf{M}^{-1}\mathbf{c}(\mathbf{x}_{jq})d\mathbf{x}_{jq} = \text{tr}(\mathbf{WM}^{-1}) \quad (15)$$

We refer to the work of Meyer and Nachtsheim (1995) for a similar expression of the  $\mathcal{V}$ -optimality criterion in the linear design setting. Finally, in terms of the inverse of the Cholesky decomposition of the information matrix (9), the average prediction variance is

$$\text{tr}(\mathbf{WM}^{-1}) = \text{tr}(\mathbf{WL}_{inv}\mathbf{L}'_{inv}). \quad (16)$$

So, to obtain the  $\mathcal{V}_B$ -optimality criterion, we have to compute  $\mathbf{W}$  for each prior parameter vector only once. The set of  $\mathbf{W}$  matrices can be re-used from one random start to the next.

### 4.3 Coordinate-exchange algorithm

The adaptive algorithm uses Meyer and Nachtsheim’s (1995) coordinate-exchange algorithm to generate Bayesian optimal designs. As opposed to the modified Fedorov algorithm employed in Kessels et al. (2006), it allows the computation of choice designs with a large number of profiles, attributes and/or attribute levels in a reasonable amount of time. The coordinate-exchange algorithm can be seen as a greedy profile exchange algorithm. Whereas the modified Fedorov algorithm possibly changes every ”coordinate” or attribute level of a profile, the coordinate-exchange algorithm only changes one coordinate. For each attribute level in the design, the coordinate-exchange algorithm tries all possible levels and chooses the level corresponding to the best value of the optimality criterion under consideration.

In contrast with the modified Fedorov algorithm, the coordinate-exchange algorithm is a candidate-set-free algorithm. That is, it does not require the specification of a set of candidate profiles. This aspect becomes more important when the candidate set is very large because of a large number of attributes and/or attribute levels. The coordinate-exchange algorithm is also substantially faster than the modified Fedorov algorithm. It runs in polynomial time, while the modified Fedorov algorithm runs in exponential time. For the comparison example, this leads to roughly a factor of three speed increase of the coordinate-exchange algorithm over the modified Fedorov algorithm. For designs with more profiles, attributes and/or attribute levels, this increase in speed becomes more pronounced.

A small disadvantage of the coordinate-exchange algorithm compared with the modified Fedorov algorithm is that it generally takes more random starts to find a globally optimal design, especially when the  $\mathcal{D}_B$ - and  $\mathcal{A}_B$ -optimality criteria are utilized. The plots in Figure 1 with estimated expected  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -efficiencies for various numbers of tries illustrate this (see also Section 3). Nevertheless, if the global optimum is not reached, the coordinate-exchange algorithm still finds a very efficient design. Also, in terms of computing time, the coordinate-exchange algorithm may be more effective than the modified Fedorov algorithm. This is certainly the case for large, realistic design problems. Therefore, the lesser performance of the coordinate-exchange algorithm per number of tries can be disregarded.

The coordinate-exchange algorithm has also been applied by Kuhfeld and Tobias (2005) to generate  $\mathcal{D}$ -efficient factorial designs for large choice experiments based on a linear model. In their SAS `%MktEx` macro, the coordinate-exchange algorithm is incorporated together with the modified Fedorov algorithm and a large catalog of orthogonal arrays. If no orthogonal design meets the design problem and the modified Fedorov algorithm is impractical to use, then the coordinate-exchange algorithm is addressed. It may also be combined with simulated annealing.

## 4.4 Small designed sample of prior parameters

In this section, we present a new method to approximate the integral related to a multivariate normal prior  $\pi(\boldsymbol{\beta}) = \mathcal{N}(\boldsymbol{\beta}|\boldsymbol{\beta}_0, \boldsymbol{\Sigma}_0)$  in the definitions of the Bayesian optimality criteria. The solution of the integral with respect to a multivariate normal prior for the various criteria has not been accomplished analytically. In general for models that are nonlinear in the parameters some numeric approximation to the integral is necessary (Chaloner and Verdinelli 1995).

Sándor and Wedel (2001) and Kessels et al. (2006) used a Monte Carlo estimate of the integral from 1,000 random draws of the prior. Such estimates are known to converge to the true value of the integral at a rate proportional to the square root of the number of draws. This necessitates a large number of draws to reduce the sample-to-sample variability to the point where different samples do not lead to different design choices. This approach is costly because the computing time for the Bayesian design is then roughly 1,000 times longer than the computing time for the local design, that is, the design for one prior parameter vector.

To solve integrals related to a multivariate normal distribution for the construction of choice designs, Sándor and Wedel (2002) utilized samples based on orthogonal arrays (Tang 1993) and Sándor and Wedel (2005) constructed quasi-Monte Carlo samples (Hickernell et al. 2000). In several cases, estimates using these methods are more efficient than Monte Carlo estimates so that it is possible to employ smaller samples to obtain the same accuracy (Sándor and András 2004; Sándor and Train 2004). There is also an extensive literature on quadrature, which is another approach to numerical integration. However, for integrals of functions in more than four dimensions, Monte Carlo estimates tend to outperform quadrature estimates (Geweke 1996; Monahan and Genz 1997).

### 4.4.1 A 20-point set

We propose to approximate the integrals in (2), (3), (4) and (6) with a designed sample of only 20 parameters. Assuming that the prior variance-covariance matrix  $\boldsymbol{\Sigma}_0$  is the identity matrix, the multivariate normal distribution is spherically symmetric around the prior mean. As a result, every parameter has the same density on a  $k$ -dimensional hypersphere of a given radius. The 20 prior parameters are uniformly distributed on such a sphere. In this way, they sample the different directions away from the prior mean fairly.

For the comparison example, the designed sample of 20 parameters yields an approximation that is worse than the Monte Carlo sample of 1,000 draws. However, in the computation of Bayesian optimal designs, it is not necessary for the approximation of the integral to be accurate. All that is required is that the sign of the difference from a rough approximation corresponding to two slightly different designs matches the sign of the difference from a better approximation. With the plot in Figure 3 we illustrate that the systematic 20-point sample and the Monte Carlo sample largely agree on design improvements in a random start.

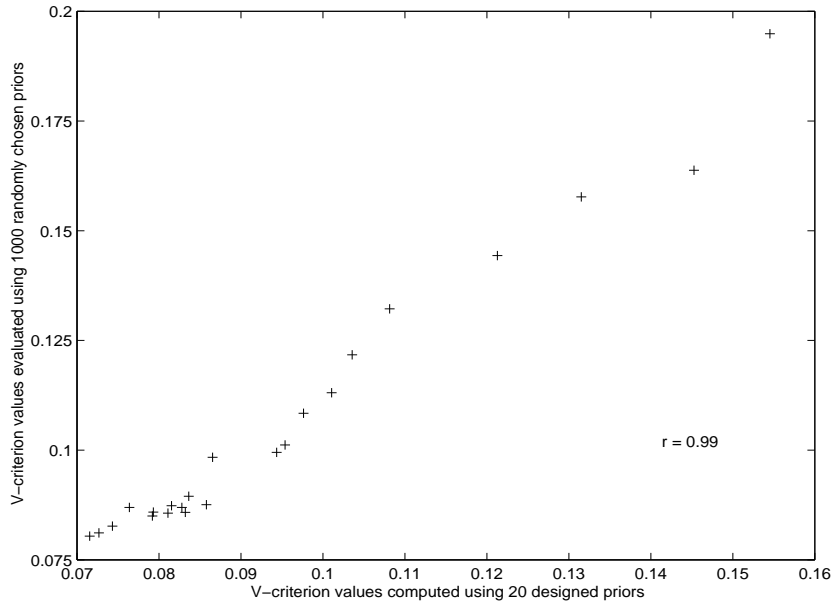


Figure 3:  $\mathcal{V}_B$ -criterion values according to the 1,000-point Monte Carlo sample versus the systematic 20-point sample and correlation between them. The points represent the course of one try of the coordinate-exchange algorithm for the two-alternative designs using the 20-point sample.

The plot compares the  $\mathcal{V}_B$ -criterion value for the Monte Carlo sample with the  $\mathcal{V}_B$ -criterion value for the systematic 20-point sample. It depicts the course of one random start of the coordinate-exchange algorithm for the two-alternative designs. A random starting design is thereby monotonically improved by making a sequence of changes, each of which improves the  $\mathcal{V}_B$ -criterion value for the systematic 20-point sample. By re-evaluating each of these changes with the  $\mathcal{V}_B$ -criterion value for the Monte Carlo sample, we find out whether every change also leads to an improvement using the better approximation.

The starting design is represented by the point at the top right of the plot, which of all points has the highest or worst  $\mathcal{V}_B$ -criterion value according to the 20-point sample as well as the Monte Carlo sample. After making one change in the original design, the second point from the top right shows an improvement in the  $\mathcal{V}_B$ -criterion value for both samples. The points proceed from the top right to the bottom left of the plot. The point at the bottom left denotes the final and best design produced in the random start. Note that this point has the lowest or best  $\mathcal{V}_B$ -criterion value as approximated by both samples. Also note that the drop in the  $\mathcal{V}_B$ -criterion value is not monotonic, indicating that the two approximations are not in complete agreement about the  $\mathcal{V}_B$ -criterion value of each change in the sequence.

Still, the agreement between the  $\mathcal{V}_B$ -criterion value for the Monte Carlo sample and the  $\mathcal{V}_B$ -criterion value for the systematic 20-point sample is clear from a correlation of 99%. Similar correlations are obtained using the coordinate-exchange algorithm with ev-

ery other design criterion and for a larger choice set size. However, this does not imply that designs that are optimal using the systematic 20-point sample are also optimal with respect to the Monte Carlo sample. The plot in Figure 4 demonstrates this.

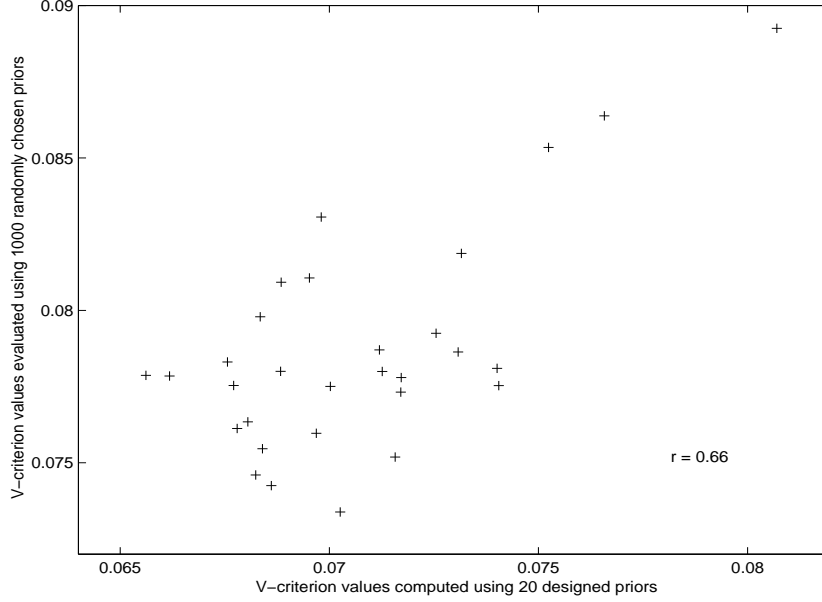


Figure 4:  $\mathcal{V}_B$ -criterion values according to the 1,000-point Monte Carlo sample versus the systematic 20-point sample and correlation between them. The points correspond to two-alternative designs produced by different tries of the coordinate-exchange algorithm using the 20-point sample.

Like the plot in Figure 3, the plot in Figure 4 displays the  $\mathcal{V}_B$ -criterion value for the Monte Carlo sample versus the  $\mathcal{V}_B$ -criterion value for the systematic 20-point sample. Now each point in the plot represents the best two-alternative design found in a single random start of the coordinate-exchange algorithm. Again, the algorithm used the  $\mathcal{V}_B$ -criterion value for the 20-point sample to generate the designs and the  $\mathcal{V}_B$ -criterion value for the Monte Carlo sample to re-evaluate them. From the plot, we see that the worst design by both  $\mathcal{V}_B$ -criterion values is the same. On the other hand, the best design according to the  $\mathcal{V}_B$ -criterion value for the 20-point sample differs from the best design indicated by the  $\mathcal{V}_B$ -criterion value for the Monte Carlo sample.

In this case, the correlation between the  $\mathcal{V}_B$ -criterion values for the Monte Carlo sample and the  $\mathcal{V}_B$ -criterion values for the 20-point sample from the different tries is only 66%. This result also applies to the other design criteria and larger choice set sizes. The fact that the correlation is not close to 100% means that it is important to check each random start using the 20-point sample with one calculation of the objective function using the Monte Carlo sample. Therefore, our approach is an adaptive one in which we re-evaluate the Bayesian designs from the 20-point sample after each try using the Monte Carlo sample. The design with the best criterion value in terms of the 1,000 draws is then selected.



Note that, if the correlation were near 100%, it would not be necessary to check the designs. On the other hand, if the correlation were not fairly large, then the adaptive approach would not work because designs using the 20-point sample would not be substantially better than random designs. Also, observe that thanks to the decrease in the number of prior parameters from 1,000 to 20 during a try we save up to 98% of the computational work!

#### 4.4.2 Constructing a small sample of prior parameters

For any choice design problem, we can construct a small set of prior parameters based on minimum potential designs or space filling designs created in JMP 6. The points of these designs are uniformly distributed on a  $k$ -dimensional hypersphere at a radius of one away from the zero vector. So on the sphere, the minimum distance to a neighboring point from any of the design points is roughly the same for all the points.

To understand how minimum potential designs are created, consider  $n$  points on a  $k$ -dimensional sphere around the zero vector. Each point,  $p$ , has levels between  $[-1, 1]$  for  $k$  continuous factors and is denoted as  $(z_{p1}, \dots, z_{pk})$ . Let  $d_{ef}$  be the distance between the  $e$ th and  $f$ th points. That is,

$$d_{ef} = \sqrt{\sum_{i=1}^k (z_{ei} - z_{fi})^2}. \quad (17)$$

The optimization problem is to find the  $n \times k$  values of  $z_{pi}$  that minimize  $E_{pot}$ , the potential energy of the system:

$$E_{pot} = \sum_{e=1}^{n-1} \sum_{f=e+1}^n \left( d_{ef}^2 + \frac{1}{d_{ef}} \right). \quad (18)$$

Here,  $d_{ef}^2$  is proportional to the energy stored in a spring when you pull it and  $1/d_{ef}$  is the potential energy between two like charged particles. When the distance between two points increases,  $d_{ef}^2$  increases. When the distance between two points decreases,  $1/d_{ef}$  increases. To visualize this, Figure 5 shows a plane with 3 design points. Each point has springs attached to the other two points. The springs pull the points together. However, each point is also positively charged and the charges repel to push the points apart. The result is that the points end up forming an equilateral triangle.

For the comparison example, the minimum potential design with 20 points in a 5-dimensional space appears in Table 3. These points lie on a sphere of a radius of one around  $[0, 0, 0, 0, 0]'$ . The minimum distance for each point to the nearest point is 1.171. If this interpoint distance seems too large, then it can be reduced by increasing the number of points.

To properly approximate the prior distribution with a 20-point sample from the points of a minimum potential design, it is necessary to rescale these points for the prior variance-covariance matrix and the prior mean. If there is no correlation between the prior coefficients or  $\Sigma_0 = \sigma_0^2 \mathbf{I}_k$ , then the 20-point sample lies on a sphere with a radius that is

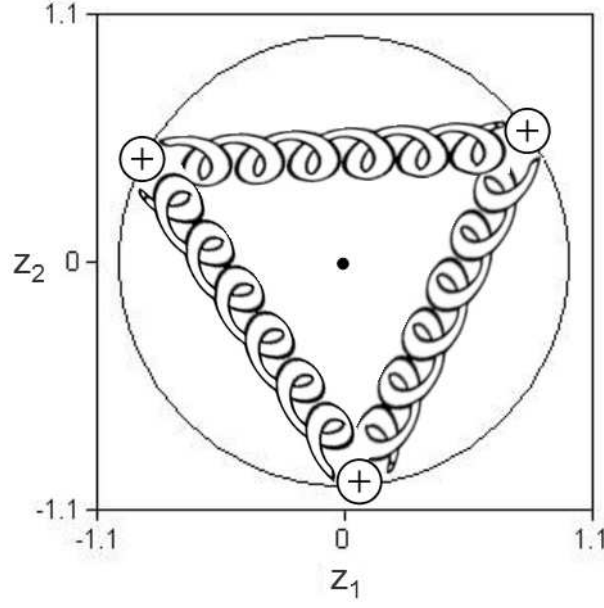


Figure 5: Three equally spaced points on the circumference of a circle.

Table 3: Minimum potential design of 20 points in 5 continuous factors for the comparison example.

Design point	$z_1$	$z_2$	$z_3$	$z_4$	$z_5$	Minimum distance	Nearest point	Radius
1	-0.17642	-0.57290	-0.19875	0.74536	-0.19600	1.17076	15	0.99281
2	-0.21775	0.81588	0.32619	-0.30104	-0.28759	1.17075	19	0.99281
3	-0.54891	-0.28739	-0.29445	0.17376	0.70655	1.17076	8	1.00000
4	-0.57116	0.06703	-0.27064	-0.77093	0.04122	1.17074	8	1.00000
5	-0.20011	-0.19572	-0.17339	-0.25973	-0.90384	1.17074	20	0.99281
6	0.00117	0.10528	0.59690	0.49371	-0.62360	1.17075	5	1.00000
7	-0.01228	0.13614	0.39319	-0.47950	0.76785	1.17076	13	0.99280
8	0.00528	-0.87552	-0.10638	-0.43810	0.15165	1.17074	4	0.99281
9	0.75353	-0.47946	-0.01214	0.10921	-0.43617	1.17076	16	1.00000
10	0.58274	0.19380	-0.32178	-0.71016	-0.08827	1.17075	20	0.99281
11	0.73699	0.47141	0.45742	0.07033	0.14296	1.17075	10	1.00000
12	-0.79511	-0.25333	0.54158	-0.02905	-0.04767	1.17077	13	0.99281
13	0.19427	-0.53359	0.65989	0.32602	0.36850	1.17075	17	1.00000
14	-0.00619	0.71761	-0.49688	-0.08192	0.48104	1.17075	2	1.00000
15	0.01039	-0.23327	-0.96643	-0.04302	-0.09815	1.17075	16	1.00000
16	0.60646	-0.18338	-0.34715	0.29484	0.61963	1.17075	15	0.99281
17	-0.19392	0.43870	0.30072	0.70409	0.42020	1.17075	13	0.99281
18	0.40102	0.49636	-0.41417	0.47335	-0.43591	1.17075	15	0.99281
19	-0.74200	0.35148	-0.35744	0.34145	-0.28555	1.17075	2	1.00000
20	0.17200	-0.17915	0.68369	-0.61868	-0.29685	1.17074	5	1.00000

proportional to the standard deviation  $\sigma_0$ . Now, the effectiveness of the 20-point sample in the adaptive algorithm depends on the radius specified, or the number of standard deviations away from the prior mean. That is to say, a well-chosen radius requires fewer random starts to reach the global optimum. To find the best radius for a spherical 20-point sample for any choice design problem, one could proceed as follows:

1. Do a number of random starts of the adaptive algorithm for each of three radii,
2. Fit a quadratic function to the minimum criterion value found at each radius,
3. Choose the radius that is the minimum of the quadratic function.

For the comparison example, we performed 10 random starts for a radius of 1, 2 and 3. Recall that  $\sigma_0 = 1$  for this example. The result for the  $\mathcal{V}_B$ -optimality criterion connected with two-alternative designs appears in Figure 6. Fitting a quadratic model to the minima results in a radius slightly larger than 2. We chose however a radius of 2 for simplicity. To illustrate the value of selecting a good radius, we compared the estimated expected efficiencies per number of tries of the two-alternative  $\mathcal{V}_B$ -optimal designs using the 20-point samples for the radii 1 and 2, respectively. The plots based on 250 tries appear in Figure 7. We clearly observe the higher expected efficiencies in case a radius of 2 is utilized as opposed to a radius of 1. We obtained similar results for any other optimality criterion in combination with any choice set size.

However, computing the "best" radius is not absolutely necessary. The heuristic of choosing a sphere radius that is twice the prior standard deviation worked well in all the examples we tried. The critical part of the adaptive algorithm is that for each random start using the 20-point sample, one checks the resulting design with the larger Monte Carlo sample. So, no matter what radius one chooses, one will have a monotonically improving set of designs as the number of random starts increases. Still, choosing a good radius increases the speed of the improvement over the random starts.

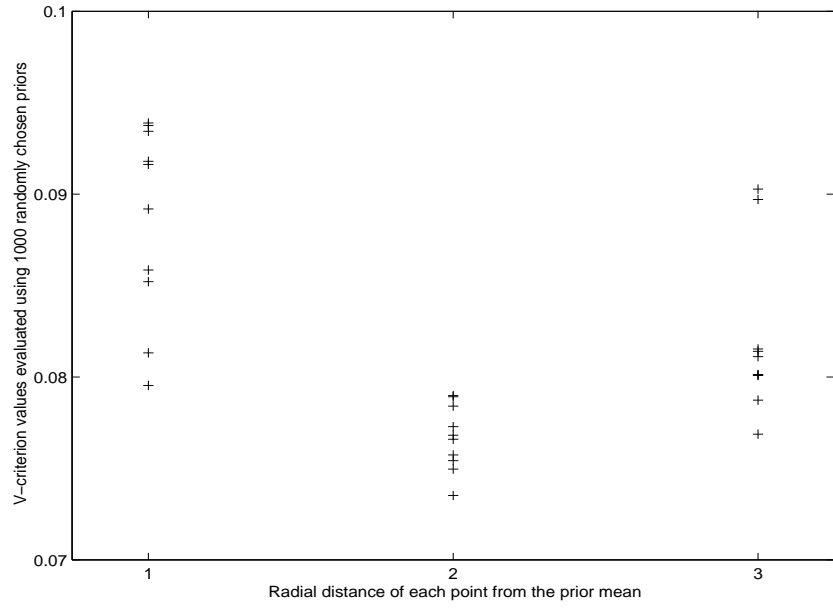


Figure 6:  $\mathcal{V}_B$ -criterion values of two-alternative designs from 10 random starts of the adaptive algorithm using the 20-point samples for the radii 1, 2 and 3.

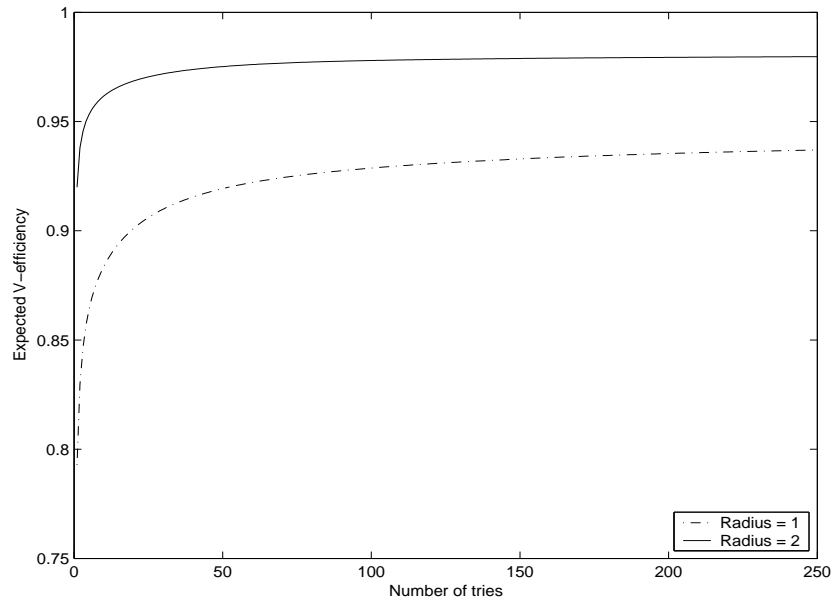


Figure 7: Estimated expected efficiencies per number of tries of the two-alternative  $\mathcal{V}_B$ -optimal designs computed using the adaptive algorithm with the 20-point samples for the radii 1 and 2.

## 5 Computation of large choice designs

The speed of the adaptive algorithm makes the computation of Bayesian optimal designs feasible for more challenging problems of larger dimensions than the rather small comparison example  $3^2 \times 2/24$ . We illustrate this with the construction of designs of two classes:  $5 \times 3 \times 2^3/2/15$  and  $5 \times 3 \times 2^3/3/10$ , jointly referred to as  $5 \times 3 \times 2^3/30$ . The designs consist of 30 profiles, grouped in 15 choice sets of size two for the first class and 10 choice sets of size three for the second class. The profiles are configured from five attributes, one of which has five levels, another of which has three levels and the three others of which have two levels. So in total, there are  $5 \times 3 \times 2^3 = 120$  candidate profiles. This candidate set is much larger than the candidate set of 18 profiles employed in the comparison example.

For the two classes, we constructed designs using the  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimality criteria. The  $\mathcal{D}_B$ -optimality criterion is the most popular criterion of the estimation-based design criteria. For the prediction-based design criteria, we prefer the  $\mathcal{V}_B$ -optimality criterion since it seeks to minimize the average prediction variance over the design region  $\chi$  and, as we showed in Section 4.2, its criterion value can be computed more efficiently than the  $\mathcal{G}_B$ -criterion value. For the two-alternative design class,  $\chi$  consists of  $Q = \binom{120}{2} = 7,140$  choice sets or 14,280 profiles, whereas for the three-alternative design class, it comprises  $Q = \binom{120}{3} = 280,840$  choice sets or 842,520 profiles. Compare these numbers with the 306 profiles and 2,448 profiles for the two- and three-alternative designs of the comparison example.

The number of parameter values,  $k$ , using effects-type coding is nine. As prior parameter distribution, we implemented the multivariate normal distribution  $\pi(\boldsymbol{\beta}) = \mathcal{N}(\boldsymbol{\beta}|\boldsymbol{\beta}_0, \boldsymbol{\Sigma}_0)$ , with  $\boldsymbol{\beta}_0 = [-1, -0.5, 0, 0.5, -1, 0, -1, -1, -1]'$  and  $\boldsymbol{\Sigma}_0 = \mathbf{I}_9$ . To obtain the designs for the  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimality criteria, we performed 1,000 tries of the adaptive algorithm for each criterion. We therefore utilized a constructed 20-point sample for the design generation and a random 1,000-point sample for the design evaluation. Again, we carried out all computations in MATLAB 7 by means of a Dell personal computer with a 1.60 GHz Intel Processor and 2 GB RAM.

The  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs for the  $5 \times 3 \times 2^3/30$  example appear in Tables A.4 and A.5 of the Appendix. Their criterion values appear in Table 4. For both optimality criteria, we notice a decrease or an improvement in the values as the choice set size goes from 2 to 3. The result that the performance in terms of prediction improves with the choice set size was also noted by Sándor and Wedel (2002) and Kessels et al. (2006). The  $\mathcal{V}_B$ -criterion values for the comparison example in Table 1 further confirm this. However, we remain undecided as to the efficiency of the  $\mathcal{D}_B$ -optimal designs with respect to the choice set size. In contrast with Table 4 where the  $\mathcal{D}_B$ -criterion values decrease with larger choice sets, the  $\mathcal{D}_B$ -criterion values in Table 1 for the comparison example increase with larger choice sets.

The computing times for one try of the adaptive algorithm to generate the two- and three-alternative  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs for the  $5 \times 3 \times 2^3/30$  example appear in

Table 4:  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -criterion values of the two- and three-alternative  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs for the  $5 \times 3 \times 2^3/30$  design example.

Optimal design	# Alternatives	
	2	3
$\mathcal{D}_B$	1.18591	1.13639
$\mathcal{V}_B$	0.23901	0.21065

Table 5. The huge design region for the three-alternative designs results in a running time of several minutes per try for the  $\mathcal{V}_B$ -optimality criterion. The computation of the  $\mathcal{V}_B$ -optimal designs for this large example would have taken months using MCMF.

Table 5: Computing times for one try of the adaptive algorithm to generate the two- and three-alternative  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs for the  $5 \times 3 \times 2^3/30$  design example. The times are expressed in hours:minutes:seconds.

Design criterion	# Alternatives	
	2	3
$\mathcal{D}_B$	00:00:08	00:00:14
$\mathcal{V}_B$	00:00:15	00:04:05

## 6 Conclusion

In this paper, we propose an adaptive algorithm for producing  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal choice designs as an alternative to the Monte Carlo modified Fedorov algorithm (MCMF) employed by Kessels et al. (2006). Kessels et al. (2006) had shown that  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs outperform  $\mathcal{D}_B$ - and  $\mathcal{A}_B$ -optimal designs for response prediction, which is central in choice experiments. However, using MCMF computing  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs is even more cumbersome than searching for  $\mathcal{D}_B$ - and  $\mathcal{A}_B$ -optimal designs so that they suggested implementing the  $\mathcal{D}_B$ -optimality criterion in practice.

Unlike MCMF, the new adaptive algorithm makes the construction of  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs practical and it allows the  $\mathcal{D}_B$ -,  $\mathcal{A}_B$ -,  $\mathcal{G}_B$ - and  $\mathcal{V}_B$ -optimal designs to be embedded in web-based conjoint choice studies with individualized designs for the respondents. We prefer using  $\mathcal{V}_B$ -optimal designs since they minimize the average prediction variance and can be computed faster than  $\mathcal{G}_B$ -optimal designs. In general, the main improvement of the adaptive algorithm over MCMF is the approximation of the normal prior distribution by a designed sample of 20 parameter vectors instead of a Monte Carlo sample of 1,000 random draws. This saves up to 98% of the computational work within each try of the algorithm. Nevertheless, we re-evaluate the designs produced by each try

using the Monte Carlo sample and adapt the design selection accordingly. This led us to call our method the adaptive algorithm.

To further speed up the design generation, the adaptive algorithm also uses a coordinate-exchange algorithm rather than a modified Fedorov algorithm. A coordinate-exchange approach saves time by avoiding the creation and use of a candidate set that grows exponentially with the number of attributes and attribute levels studied. Thus, the time savings of the coordinate-exchange algorithm increase with the number of profiles, attributes and attribute levels. As a last way to accelerate the computations for any optimality criterion, the adaptive algorithm incorporates an update formula to economically calculate the optimality criterion values of designs.

The computational speed of the adaptive algorithm makes the use of individualized Bayesian optimal designs in web-based surveys possible. To examine what is the best way to do this, is beyond the scope of this paper. We expect, however, that such an approach would allow an efficient estimation of mixed logit (Sándor and Wedel 2002) and latent class models (Andrews et al. 2002; Train 2003) that aim at modelling consumer heterogeneity. Another topic for further research is the construction of designs for choice experiments in which one suspects correlation between parameter coefficients. In that case, the multivariate normal prior distribution is elliptically symmetric around the prior mean. The small designed sample of parameters from a minimum potential design should then be rescaled to lie on a  $k$ -dimensional ellipsoid. Lastly, the efficiency of optimal designs with respect to the choice set size might be further investigated.

## Acknowledgements

We would like to thank Prof. Hans Nyquist for hosting the first author for six weeks at the Department of Statistics of the University of Stockholm, Sweden, while she was working on this paper.



## Appendix. Choice design tables

Table A.1: Two-alternative Bayesian optimal designs for the  $3^2 \times 2/24$  example.

Choice set	Alt	$\mathcal{D}_B$			$\mathcal{A}_B$			$\mathcal{G}_B$			$\mathcal{V}_B$		
		Attr			Attr			Attr			Attr		
		1	2	3	1	2	3	1	2	3	1	2	3
1	I	1	2	2	2	1	2	1	2	2	1	3	2
	II	2	1	1	1	2	1	3	1	2	2	1	2
2	I	3	1	1	1	2	2	1	2	1	1	2	2
	II	2	3	2	2	1	1	2	3	1	2	3	2
3	I	2	3	2	2	2	2	2	1	1	2	2	2
	II	1	2	2	1	1	1	3	2	2	1	1	2
4	I	3	3	2	2	2	1	2	2	1	2	2	1
	II	2	2	1	3	1	1	1	3	1	1	3	2
5	I	2	2	2	2	1	1	1	2	1	1	1	1
	II	1	3	2	1	2	1	2	1	2	2	2	2
6	I	1	2	2	2	3	2	3	3	1	2	1	1
	II	3	1	2	3	2	1	2	1	1	3	1	2
7	I	3	1	2	1	2	2	2	3	2	1	2	2
	II	1	3	1	2	3	2	3	1	1	3	1	2
8	I	1	2	1	1	3	1	3	2	1	1	1	1
	II	2	1	2	3	1	2	1	1	1	3	3	2
9	I	2	2	2	1	1	1	3	1	2	1	2	1
	II	1	1	1	3	1	1	1	3	1	2	1	2
10	I	2	2	1	3	3	2	2	2	1	2	3	2
	II	1	1	1	2	1	1	3	3	2	3	1	1
11	I	3	2	1	2	2	2	2	2	2	3	2	2
	II	2	1	1	1	3	2	1	1	1	2	1	2
12	I	2	1	1	2	2	1	1	1	2	1	2	2
	II	3	2	2	1	1	1	2	1	1	1	3	1

Table A.2: Three-alternative Bayesian optimal designs for the  $3^2 \times 2/24$  example.

Choice set	Alt	$\mathcal{D}_B$			$\mathcal{A}_B$			$\mathcal{G}_B$			$\mathcal{V}_B$		
		Attr			Attr			Attr			Attr		
		1	2	3	1	2	3	1	2	3	1	2	3
1	I	2	1	1	1	2	1	2	1	1	3	1	1
	II	1	3	1	2	1	2	1	2	2	1	3	2
	III	1	2	2	1	3	2	1	3	1	2	2	1
2	I	2	3	2	1	1	1	2	2	1	3	1	1
	II	1	2	1	1	2	2	1	3	2	2	2	2
	III	3	1	2	2	2	1	3	2	2	3	2	2
3	I	1	2	1	2	2	2	2	3	1	3	1	2
	II	2	3	1	1	2	2	2	1	2	2	3	2
	III	1	3	2	1	1	1	3	3	1	2	2	1
4	I	3	2	2	1	2	1	1	1	1	1	3	1
	II	2	1	1	1	1	1	3	1	2	2	1	1
	III	1	3	2	2	1	1	2	3	2	1	2	2
5	I	2	1	2	2	2	1	3	2	1	1	1	1
	II	2	2	2	3	1	1	3	3	1	2	3	2
	III	1	1	1	1	3	1	1	3	1	3	2	2
6	I	3	3	2	1	2	1	1	2	1	3	3	2
	II	2	2	1	2	3	1	2	3	2	1	3	1
	III	1	1	1	3	3	2	2	3	1	2	1	2
7	I	3	1	2	2	3	2	1	1	1	1	3	2
	II	2	1	1	3	1	1	1	2	2	2	3	2
	III	3	2	1	3	3	1	2	1	2	1	2	2
8	I	1	2	1	2	1	1	2	1	1	3	2	1
	II	3	1	1	3	2	1	1	1	2	3	1	2
	III	2	2	2	1	2	1	2	2	1	1	1	1

Table A.3: Four-alternative Bayesian optimal designs for the  $3^2 \times 2/24$  example.

Choice set	Alt	$\mathcal{D}_B$			$\mathcal{A}_B$			$\mathcal{G}_B$			$\mathcal{V}_B$		
		Attr			Attr			Attr			Attr		
		1	2	3	1	2	3	1	2	3	1	2	3
1	I	2	2	2	2	3	2	2	3	1	3	1	1
	II	2	1	1	3	2	2	2	2	1	1	2	1
	III	3	1	2	2	2	2	1	3	1	2	1	1
	IV	3	2	1	1	1	1	1	2	2	3	2	2
2	I	2	1	2	1	3	1	3	2	1	3	2	2
	II	3	1	1	2	3	2	2	1	2	1	1	1
	III	3	2	2	1	3	2	1	1	1	2	3	2
	IV	1	2	1	2	1	1	3	1	1	2	2	2
3	I	2	2	1	3	1	1	1	3	2	2	3	1
	II	1	1	1	3	2	1	1	1	2	1	3	2
	III	1	3	2	1	2	2	1	2	1	2	1	2
	IV	3	1	2	2	1	1	2	2	2	1	2	1
4	I	1	2	1	2	2	1	2	1	1	3	3	1
	II	2	1	1	1	1	1	3	1	2	3	1	1
	III	1	3	1	3	1	2	1	2	2	1	3	1
	IV	2	3	2	1	3	2	2	3	2	2	2	2
5	I	3	2	2	3	2	2	2	3	1	2	2	1
	II	2	2	2	1	2	1	3	3	1	2	1	1
	III	2	3	2	3	1	1	3	1	1	1	3	2
	IV	1	1	1	1	1	1	3	3	2	3	1	2
6	I	1	2	2	1	3	1	3	1	2	2	3	2
	II	2	3	1	2	1	2	1	2	2	2	2	1
	III	1	3	1	2	3	1	2	1	2	1	2	2
	IV	2	1	1	1	2	1	1	1	1	1	3	1

Table A.4: Two-alternative  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs for the  $5 \times 3 \times 2^3/30$  example.

Choice set	Alt	$\mathcal{D}_B$					$\mathcal{V}_B$				
		Attr					Attr				
		1	2	3	4	5	1	2	3	4	5
1	I	5	2	1	2	2	4	1	1	2	2
	II	4	3	2	1	2	5	1	2	2	1
2	I	3	2	2	1	1	2	1	1	1	2
	II	2	3	1	2	1	4	2	2	1	2
3	I	5	1	2	1	1	3	3	1	2	1
	II	1	2	1	2	1	5	2	1	2	1
4	I	4	1	1	1	1	4	3	1	1	2
	II	2	2	1	1	1	3	2	1	1	2
5	I	3	1	1	2	1	5	3	2	2	1
	II	4	2	2	1	2	2	2	2	1	1
6	I	1	2	2	1	1	2	1	2	2	1
	II	2	1	2	1	1	3	2	2	2	2
7	I	2	2	2	2	1	3	1	2	2	1
	II	3	1	1	2	2	2	3	2	1	2
8	I	1	1	1	1	1	1	3	1	2	2
	II	4	3	1	2	2	3	3	1	2	2
9	I	2	1	1	1	2	2	3	2	2	1
	II	5	2	2	2	1	1	1	2	1	1
10	I	5	3	1	1	2	4	3	2	2	2
	II	4	2	2	1	1	3	3	2	1	2
11	I	1	1	2	2	2	2	2	2	1	1
	II	3	2	2	2	2	5	1	1	1	2
12	I	3	1	2	1	1	1	2	2	1	1
	II	5	1	2	1	2	2	2	2	1	2
13	I	1	3	2	1	1	4	1	2	2	1
	II	3	2	1	2	1	1	2	1	2	1
14	I	4	1	2	2	1	4	1	2	1	1
	II	3	3	2	2	2	1	3	2	2	1
15	I	3	3	2	2	1	5	3	2	1	2
	II	4	2	1	1	2	4	2	1	2	1

Table A.5: Three-alternative  $\mathcal{D}_B$ - and  $\mathcal{V}_B$ -optimal designs for the  $5 \times 3 \times 2^3/30$  example.

Choice set	Alt	$\mathcal{D}_B$					$\mathcal{V}_B$				
		Attr					Attr				
		1	2	3	4	5	1	2	3	4	5
1	I	4	3	1	2	2	5	1	2	1	1
	II	1	1	1	1	1	4	3	2	1	2
	III	2	2	2	1	1	3	2	2	1	2
2	I	2	3	2	1	2	1	1	1	2	1
	II	1	3	1	2	1	5	2	2	1	2
	III	5	2	2	1	2	3	3	2	1	2
3	I	4	3	2	1	1	1	2	2	1	2
	II	3	3	1	1	2	3	2	2	2	1
	III	2	2	1	1	1	4	1	2	2	1
4	I	2	3	1	1	1	4	2	1	1	1
	II	1	2	1	1	1	2	3	1	2	1
	III	5	3	1	2	2	1	1	1	1	1
5	I	3	1	2	2	1	5	3	2	1	2
	II	1	2	2	2	1	1	3	1	1	1
	III	2	2	2	2	2	4	2	1	2	1
6	I	2	1	2	2	1	3	3	2	1	1
	II	3	2	2	1	2	2	2	1	1	2
	III	4	1	2	1	2	5	3	1	1	2
7	I	4	2	1	2	2	2	2	1	1	2
	II	3	1	2	1	1	5	3	2	2	1
	III	5	3	2	2	1	4	2	2	1	2
8	I	4	1	1	2	1	1	2	1	2	1
	II	2	3	1	1	2	3	1	1	1	2
	III	3	3	2	2	1	2	1	2	1	1
9	I	4	2	2	2	1	5	1	2	2	1
	II	1	3	2	2	1	2	3	2	1	1
	III	5	1	1	2	2	3	3	1	2	2
10	I	3	2	1	2	1	1	3	2	1	2
	II	5	1	2	1	1	2	3	1	2	2
	III	1	3	1	1	2	3	1	1	1	2

## References

- Andrews, R. L., Ainslie, A. and Currim, I. S. (2002). An empirical comparison of logit choice models with discrete versus continuous representations of heterogeneity, *Journal of Marketing Research* **39**: 479–487.
- Bennett, J. M. (1965). Triangular factors of modified matrices, *Numerische Mathematik* **7**: 217–221.
- Chaloner, K. and Verdinelli, I. (1995). Bayesian experimental design: a review, *Statistical Science* **10**: 273–304.
- Cook, R. D. and Nachtsheim, C. J. (1980). A comparison of algorithms for constructing exact  $\mathcal{D}$ -optimal designs, *Technometrics* **22**: 315–324.
- Fedorov, V. V. (1972). *Theory of optimal experiments*, New York: Academic Press.
- Geweke, J. (1996). Monte Carlo simulation and numerical integration, in *Handbook of Computational Economics*, Vol. 1, Ch. 15, Amman H. M., Kendrick D. A., Rust J., eds. Amsterdam: Elsevier Science, 731–800.
- Hickernell, F. J., Hong, H. S., L’Ecuyer, P. and Lemieux, C. (2000). Extensible lattice sequences for quasi-Monte Carlo quadrature, *SIAM Journal on Scientific Computing* **22**: 1117–1138.
- Huber, J. and Zwerina, K. (1996). The importance of utility balance in efficient choice designs, *Journal of Marketing Research* **33**: 307–317.
- Kessels, R., Goos, P. and Vandebroek, M. (2006). A comparison of criteria to design efficient choice experiments, *Journal of Marketing Research* **43**: 409–419.
- Kuhfeld, W. F. and Tobias, R. D. (2005). Large factorial designs for product engineering and marketing research applications, *Technometrics* **47**: 132–141.
- Meyer, R. K. and Nachtsheim, C. J. (1995). The coordinate-exchange algorithm for constructing exact optimal experimental designs, *Technometrics* **37**: 60–69.
- Monahan, J. and Genz, A. (1997). Spherical-radial integration rules for Bayesian computation, *Journal of the American Statistical Association* **92**: 664–674.
- Sándor, Z. and András, P. (2004). Alternative sampling methods for estimating multivariate normal probabilities, *Journal of Econometrics* **120**: 207–234.

- Sándor, Z. and Train, K. (2004). Quasi-random simulation of discrete choice models, *Transportation Research B* **38**: 313–327.
- Sándor, Z. and Wedel, M. (2001). Designing conjoint choice experiments using managers' prior beliefs, *Journal of Marketing Research* **38**: 430–444.
- Sándor, Z. and Wedel, M. (2002). Profile construction in experimental choice designs for mixed logit models, *Marketing Science* **21**: 455–475.
- Sándor, Z. and Wedel, M. (2005). Heterogeneous conjoint choice designs, *Journal of Marketing Research* **42**: 210–218.
- Tang, B. (1993). Orthogonal array-based Latin hypercubes, *Journal of the American Statistical Association* **88**: 1392–1397.
- Train, K. E. (2003). *Discrete Choice Methods with Simulation*, Cambridge U.K.: Cambridge University Press.